REMARKS

Claims 1-29 are pending in the application and are presented for reconsideration. The remaining independent claims are claims 1, 19 and 24. By the foregoing amendments, claims 1, 19 and 24 are sought to be amended.

These changes are believed not to introduce new matter, and their entry is respectfully requested. The claims have been amended merely to clarify the claims and expedite the prosecution of the application, not to overcome any cited references.

Based on the above Amendment and the following Remarks, Applicants respectfully requests that the Examiner reconsider all outstanding objections and rejections, and withdraw them.

The Examiner has rejected claims 1-29 under the obviousness provisions of 35 U.S.C. §103 as allegedly being unpatentable over U.S. Patent Publication No. 2002/0133412 to Oliver (hereafter referred to as "Oliver") in view of US Patent Publication No. 2003/0197719 to Lincke et al. (hereafter referred to as "Lincke") and further in view of US Patent 7,089,249 to Kobayashi et al (hereafter referred to as "Kobayashi"). Based on the above Amendment and the following Remarks, Applicants respectfully requests that the Examiner reconsider the rejection, and withdraw it.

As amended, independent claims 1, 19 and 24 recite:

1. A computer implemented method for dynamically rendering data in a markup language, the method comprising:
identifying a symbol in the data in the markup language, the symbol indicating a query of a data set;
accessing the data set in order to generate a resolution to the query; and
dynamically rendering the resolution to the query as a part of the markup language, according to at least one rule associated with the markup language wherein said symbol can be used to dynamically render multiple data sets.

19. A computer program product for dynamically rendering data in a markup language, the computer program product comprising:
program code for identifying a symbol in the data in the markup language, the symbol indicating a query of a data set;
program code accessing the data set in order to generate a resolution to the query;
program code for dynamically rendering the resolution to the query as a part of the markup language, according to at least one rule associated with the markup language wherein said symbol can be used to dynamically render multiple data sets; and
a computer readable medium on which the program codes are stored.

24. A computer system for dynamically rendering data in a markup language, the computer system comprising:
an identification module, for identifying a symbol in the data in the markup language, the symbol indicating a query of a data set;
a data access module, for accessing the data set in order to generate a resolution to the query, the data access module being coupled to the identification module; and
a rendering module, for dynamically rendering the resolution to the query as a part of the markup language, according to at least one rule associated with the markup language wherein said symbol can be used to dynamically render multiple data sets, the rendering module being coupled to the data access module.

The primary reference (Oliver) discloses a system for management of transactions on networks. Oliver defines Token Validation Service (TVS) and user access management environment. It addresses a distributed service provider environment where users belong to one of an available service provider. The Oliver system allows the service providers to give service to users on other providers, thus sharing users without the need of creating a user

account explicitly. Oliver describes a billing procedure that each service provider will use to charge the users of other providers for its service. The system also provides client authenticity verification by Token Validation Service (TVS). TVS is implemented using object oriented client/server approach. The clients of the TVS service are the content providers (Currently HTTP content providers) who can link each other and provide their user base with more services.

In contrast to Oliver, the present invention can categorize approaches for rendering data in a markup language in two ways, static and dynamic. Static methods are conventional and is what is described in Oliver. In a static method the query description is given in the exact form which needs to be identified, accessed, matched and rendered in its exact pattern structure. The rendered screen has a direct structural correspondence to the query. This is simple to implement and is well known in conventional systems such as Oliver. Static methods have several shortcomings. These shortcomings are overcome by the dynamic approach of the claimed invention where the query is more generic, is capable of being matched more generally to multiple data sets and can produce a variety of results which need to be rendered in different forms as expressed in the query. The static methods cannot achieve this. Moreover, the methods/ programs/ systems for (i) identifying a symbol in the data in the markup language, the symbol indicating a query of the data set, (ii) accessing the data set in order to generate a resolution to the query and (iii) rendering the resolution to the query as a part of the markup language, according to at least one rule associated with the markup language, are different for static and dynamic approaches. The advantages of the dynamic approach in having smaller queries, smaller databases, allowing for client side modeling as described in the patent application are absent in the static approach known.

Oliver describes an end user "query" to the service providers through HTML forms or otherwise (see paragraph [0275] page 11). However the context and use of the term "query" is different than in the present invention. The query generation in Oliver et al ([0346] Page 15) essentially requests an html page from the server. Oliver is silent toward, and therefore does not teach or suggest, dynamically rendering the content. In contrast all of the independent claims require dynamic rendering.

In the office action Lincke is used to teach dynamic rendering. However, each of the independent claims recite the feature of "wherein said symbol can be used to dynamically render multiple data sets" wherein this query symbol is embedded into the markup language such that the semantics of the information are preserved and the information can be used in multiple places and form multiple data sets in a single application. There is no teaching or suggestion of this from Lincke. The office action asserts that Kobayashi at column 9, lines 52-67 teaches the feature of "wherein said symbol can be used to render multiple data sets." Applicants have reviewed Kobayashi and do not see where Kobayashi teaches a dynamic rendering where a symbol can be used to dynamically render multiple data sets.

Accordingly, Applicants request that the Examiner reconsider and withdraw the rejection of claims 1, 19 and 24 along with all claims dependent thereon.

Conclusion

Applicants believe that all of the stated grounds of objection and rejection set forth by the Examiner in the Office Action have been properly accommodated or addressed. Applicants, therefore, respectfully request that the Examiner reconsider all presently outstanding objections and rejections and withdraw them. The Examiner is invited to telephone the undersigned representative if it is felt that an interview might be useful for any reason.

Respectfully submitted

Date: 20 Feb 2007

By: /John T. McNelis/

John T. McNelis
Attorney for Applicants
Reg. No. 37,186
FENWICK & WEST LLP
Silicon Valley Center
801 California Street
Mountain View, CA 94306
(650) 335-7133
jmcnelis@fenwick.com